APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/972,076	10/05/2001	Gregory A. Johnson	ISAA0004	4558

22862　　　7590　　　04/21/2005

GLENN PATENT GROUP
3475 EDISON WAY, SUITE L
MENLO PARK, CA 94025

EXAMINER
PHAM, CHRYSTINE

ART UNIT	PAPER NUMBER
2192	

DATE MAILED: 04/21/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE <u>3</u> MONTH(S) FROM
THE MAILING DATE OF THIS COMMUNICATION.
- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed
  after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
  Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any
  earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *24 November 2004*.

2a)☐ This action is **FINAL**.   2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is
closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-47,49-70 and 72-79* is/are pending in the application.

　4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-47,49-70 and 72-79* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

　Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

　Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

　a)☐ All　b)☐ Some *　c)☐ None of:

　　1.☐ Certified copies of the priority documents have been received.

　　2.☐ Certified copies of the priority documents have been received in Application No. _____.

　　3.☐ Copies of the certified copies of the priority documents have been received in this National Stage
application from the International Bureau (PCT Rule 17.2(a)).

　* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1)☒ Notice of References Cited (PTO-892)

2)☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3)☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
　Paper No(s)/Mail Date _____.

4)☐ Interview Summary (PTO-413)
　Paper No(s)/Mail Date. _____.

5)☐ Notice of Informal Patent Application (PTO-152)

6)☐ Other: _____.

DETAILED ACTION

1.   This action is responsive to the Amendments filed on 11/24/2004. Claims 1, 9, 17, 18, 26, 32, 33,

34, 37, 38, 39, 53, 56, 57, 60, 61, 62, 76, 78, 79 have been amended. Claims 48 and 71 have

been cancelled. Claims 1-47, 49-70, 72-79 are presented for examination.

2.   Claims 48 and 71 are now being rejected under new applicable arts discovered.

Response to Amendment

3.   In view of the amendments made to claims 32, 33, 37, 38, 39, 53-55, 56, 60, 61, 62, 76-78, and

79 in response to 35 U.S.C. 112 rejections of the claims, rejections to claims are hereby

withdrawn. Claims 46 and 69, which have also been rejected under 35 U.S.C. 112, have not yet

been amended, therefore, rejection under 35 U.S.C. 112 is maintained for these claims.

Claim Rejections - 35 USC § 112

4.   The following is a quotation of the first paragraph of 35 U.S.C. 112:

*The specification shall contain a written description of the invention, and of the manner and process of
making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to
which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the
best mode contemplated by the inventor of carrying out his invention.*

5.   Claims 1-79 are rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the

enablement requirement. The claim(s) contains subject matter which was not described in the

specification in such a way as to enable one skilled in the art to which it pertains, or with which it

is most nearly connected, to make and/or use the invention.

        Claims 1, 9, 17, 18, 26, 34, and 57 contain the subject matter "champion/challenger"

experiment for testing a new strategy during a strategy design cycle. This subject matter of

champion/challenger testing [a strategy or rule] was not described in the specification in such a

way as to enable one skilled in the art to which it pertains, or with which it is most nearly

connected, to make and/or use the invention. On page 27 (lines 15-20), the Specification

discloses "champion/challenger testing" as comparing competing strategies or rules to determine which strategy or rule produces the best results. And the comparison seems to be based on each strategy's "effectiveness", that is to say, whichever strategy is proven to be "more effective", then it becomes the "champion" strategy. However, it is not clear how "effectiveness" [of the strategy/rule] is being defined. On page 22 (lines 4-12), the Specification mentions "effectiveness of rules" as being related to statistical tracking. Monitoring for hits (i.e., the number of times a strategy/rule is used) is also discussed in the same paragraph. However, it is not clear whether "effectiveness" is in fact being defined as the number of hits (the number of times a strategy is used). Assuming, arguendo, that it is, it still raises doubts as to how reliable, accurate, and enabling, this testing method is since the disclosed method of measuring "effectiveness" does not seem to utilize any definite time frame as a base standard for measuring hits (e.g., how many minutes/hours/days/weeks/months/years should the competing strategies be monitored for hits before a decision can be made as to which strategy is "more effective"?). Lacking a standard time frame, test results cannot be reliable or accurate.

Claims 2-8, 10-16, 19-25, 27-33, 35-56, 58-79 are also rejected under 35 U.S.C. 112 as dependent claims of rejected base claims.

6.   The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

Claims 46 and 69 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

A broad range or limitation together with a narrow range or limitation that falls within the broad range or limitation (in the same claim) is considered indefinite, since the resulting claim does not clearly set forth the metes and bounds of the patent protection desired. Note the explanation given by the Board of Patent Appeals and Interferences in *Ex parte Wu*, 10

USPQ2d 2031, 2033 (Bd. Pat. App. & Inter. 1989), as to where broad language is followed by

"such as" and then narrow language. The Board stated that this can render a claim indefinite by

raising a question or doubt as to whether the feature introduced by such language is (a) merely

exemplary of the remainder of the claim, and therefore not required, or (b) a required feature of

the claims. Note also, for example, the decisions of *Ex parte Steigewald*, 131 USPQ 74 (Bd. App.

1961); *Ex parte Hall*, 83 USPQ 38 (Bd. App. 1948); and *Ex parte Hasche*, 86 USPQ 481 (Bd.

App. 1949). For example, claim 46 recites the broad recitation "said decisions comprise any

combination of, but are not limited to:" (line 3), and the claim also recites "scores; reason codes;

actions; and other calculated results" (line 4-7) which is the narrower statement of the

range/limitation. Claim 69 also recites the broad recitation "said decisions comprise any

combination of, but are not limited to:" (line 3), and the claim also recites "scores; reason codes;

actions; and other calculated results" (line 4-7) which is the narrower statement of the

range/limitation.

Claim Rejections - 35 USC § 103

7.   The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness

rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

8.   Claims 1, 2, 4-5, 8-9, 10, 12-13, 16-17, 34-38, 40-47, 49, 52, 57-61, 63-72, 75 are rejected under

35 U.S.C. 102(b) as being unpatentable by Courts et al. (*Courts et al.,* US 6085220), in view of

Harrison et al. (*Harrison et al.,* US 6741976 B1).

        As per claim 1, *Courts et al.* teach an apparatus (e.g., see Abstract, *enterprise interaction

hub 10* FIG.1 & associated text) and method (e.g., col.1:45-49) for an all-purpose decision

service/server/engine returning a real-time decision in ASP mode to an end user/client (e.g.,
col.3:24-27 & 34-35, col.7:38-46, col.9:30-35), said method comprising:

- o linking to a first computer system having project design software (e.g., see *business layer 16* FIG.1 & associated text) via the Internet or a virtual private network (e.g., see *DCOM* FIG.1) for designing rules, models, and/or strategies (e.g., see *business logic & business object 20* FIG.1 & associated text);

- o passing control (e.g., see *integration layer 18* FIG.1 & associated text) to a code generator server (e.g., see *Independent Software Vendor ISV space 28* FIG.1 & associated text) for generating code for use in production in said ASP environment (e.g., col.2:62-67);

- o said code generator server generating strategy service software for installation on a decision engine/server which is embeddable in a software application (e.g., see *presentation layer 14 & render object/engines 20* FIG.1, see *render engines 122* FIG.2 & associated text, see *Rengine.PBD 130 & application code PBDs 132* FIG.2B & associated text) for executing said rules/models (e.g., col.1:56-58, col.6:52-54);

- o sending input data to said decision server via a Web server (e.g., see Abstract, see *interaction layer 12 & HTTP* FIG.1 & associated text, col.9:30-32), said input data for processing using said decision server;

- o said decision server processing said input data according to said installed rules, models (e.g., col.1:56-58, col.3:51-52) and creating corresponding output data/calculated results/actions (e.g., see *html generation* FIG.1 & associated text);

- o said decision server returning said created output data to said Web server in XML format (e.g., col.4:13-16); and

- o said Web server returning said output data (e.g., see Abstract, col.1:52-54).

- o a transaction log of said automated real time decisions, said log accessible by a client (e.g., col. 4:39-40 & 47-48).

Courts et al. disclose wherein said project design software further comprises capability for inserting an experiment for testing a new strategy (e.g., col.4:7-9; col.5:10-13; *business layer 16, business objects 22* FIG.1 & associated text). However, *Courts et al.* do not expressly disclose said experiment as champion/challenger experiment. However, *Harrison et al.* disclose a project design software (see at least *software development, solutions, learning classifier-system* col.1:25-60) comprises capability for inserting a champion/challenger experiment for testing a new strategy (see at least *learning classifier-system, rules, fitness* col.1:49-58; *genetic programming, single solution, learning classifier systems, multiple actions, knowledge base, optimum behavior, diverse rule base* col.2:4-20; *intelligent agents, rules, associated fitness, success measure* col.2:39-col.3:10; *FEHN, new rule, existing rule, effectiveness* col.3:45-50; *Engine 160, rules, survival of the fittest* col.7:20-36; *winning rule* col.11:60-67; *agent, previous best agent* col.26:35-46). It would have been obvious to one of ordinary skill in the pertinent art at the time the invention was made to incorporate the teaching of *Harrison et al.* into that of *Courts et al.* for the inclusion of champion/challenger experiment for testing a new strategy. And the motivation for doing so would have been to enable automatic and continual optimization or online adaptation (i.e., improving performance) of the software to minimize down time and sub-optimal functioning while awaiting new software version and avoid the ongoing needs to understand the software, modify tests and integrate the modified software (see at least col.2:20-36; col.4:17-33).

As per claim 2, *Courts et al.* teach the method as applied to Claim 1, further comprising:

o   using system integration (e.g., see *integration layer 18* FIG.1 & associated text) and consulting services (e.g., see *trend DB 36 & profile DB 38 & enterprise space 26* FIG.1 & associated text), said consulting services for developing and refining rules, models, and strategies (e.g., col.1:61-col.2:4, col.3:3-8).

As per claim 4, *Courts et al.* teach the method as applied to Claim 1, wherein said decision server is linked to external data resources for extracting additional relevant data (e.g., see *profile DB 38 & enterprise space 26* FIG.1 & associated text).

As per claim 5, *Courts et al.* teach the method as applied to claim 1 wherein an ASP file

running on a Web server passes input data to said decision server (e.g., col.3:18-24), said input

data is in XML format (e.g., col.4:13-16).

As per claim 9, *Courts et al.* teach an apparatus for a decision service returning a real-

time decision in ASP mode to an end user (see claim 1), said apparatus comprising:

- o   means for linking to a first computer system having project design software (e.g., see

    business layer 16 FIG.1 & associated text) via the Internet or a virtual private network

    (e.g., see *DCOM* FIG.1) for designing rules/models (e.g., see *business logic* &

    business object 20 FIG.1 & associated text);

- o   means for passing control (e.g., see *integration layer 18* FIG.1 & associated text) to a

    code generator server (e.g., see *Independent Software Vendor ISV space 28* FIG.1

    & associated text) for generating code for use in production in said ASP environment

    (e.g., col.2:62-67);

- o   means for said code generator server generating strategy service software for

    installation on a decision server (e.g., see *presentation layer 14 & render

    object/engines 20* FIG.1, see *render engines 122* FIG.2 & associated text, see

    Rengine.PBD 130 & application code PBDs 132 FIG.2B & associated text) for

    executing said rules/models (e.g., col.1:56-58, col.6:52-54);

- o   means for sending input data to said decision server via a Web server (e.g., see

    Abstract, see *interaction layer 12 & HTTP* FIG.1 & associated text, col.9:30-32), said

    input data for processing using said decision server;

- o   means for said decision server processing said input data according to said installed

    rules/models (e.g., col.1:56-58, col.3:51-52) and creating corresponding output data

    (e.g., see *html generation* FIG.1 & associated text);

   o means for said decision server returning said created output data to said Web server in XML format (e.g., col.4:13-16); and

   o means for said Web server returning said output data (e.g., see Abstract, col.1:52-54).

Courts et al. disclose wherein said project design software further comprises capability for inserting an experiment for testing a new strategy (e.g., col.4:7-9; col.5:10-13; *business layer 16, business objects 22* FIG.1 & associated text). However, *Courts et al.* do not expressly disclose said experiment as champion/challenger experiment. However, *Harrison et al.* disclose a project design software (see at least *software development, solutions, learning classifier-system* col.1:25-60) comprises capability for inserting a champion/challenger experiment for testing a new strategy (see at least *learning classifier-system, rules, fitness* col.1:49-58; *genetic programming, single solution, learning classifier systems, multiple actions, knowledge base, optimum behavior, diverse rule base* col.2:4-20; *intelligent agents, rules, associated fitness, success measure* col.2:39-col.3:10; *FEHN, new rule, existing rule, effectiveness* col.3:45-50; *Engine 160, rules, survival of the fittest* col.7:20-36; *winning rule* col.11:60-67; *agent, previous best agent* col.26:35-46). It would have been obvious to one of ordinary skill in the pertinent art at the time the invention was made to incorporate the teaching of *Harrison et al.* into that of *Courts et al.* for the inclusion of champion/challenger experiment for testing a new strategy. And the motivation for doing so would have been to enable automatic and continual optimization or online adaptation (i.e., improving performance) of the software to minimize down time and sub-optimal functioning while awaiting new software version and avoid the ongoing needs to understand the software, modify tests and integrate the modified software (see at least col.2:20-36; col.4:17-33).

   As per claims 8, 10, 12, and 13, 16, they recite limitations, which have been addressed in claims 1, 2, 4, and 5 respectively, therefore, are rejected for the same reasons as cited in claims 1, 2, 4, and 5.

As per claim 17, *Courts et al.* teach a method for assembling and delivering an all-purpose decision engine/server in ASP mode, said method comprising:

- o   defining input and output structures in XML format (e.g., see claim 5).

- o   importing analytical models (e.g., see claim 1).

- o   adding rules, modifying decision actions (e.g., col.5:10-13), and general tweaking of said engine (e.g., see _business layer 16, business objects 22 FIG.1 & associated text).

- o   testing [rules within] said engine (e.g., col.4:7-9).

- o   fueling said engine with data from a variety of sources and said engine delivering decisions (see claim 4).

Courts et al. disclose wherein said project design software further comprises capability for inserting an experiment for testing a new strategy (e.g., col.4:7-9; col.5:10-13; *business layer 16, business objects 22* FIG.1 & associated text). However, *Courts et al.* do not expressly disclose said experiment as champion/challenger experiment. However, *Harrison et al.* disclose a project design software (see at least *software development, solutions, learning classifier-system* col.1:25-60) comprises capability for inserting a champion/challenger experiment for testing a new strategy (see at least *learning classifier-system, rules, fitness* col.1:49-58; *genetic programming, single solution, learning classifier systems, multiple actions, knowledge base, optimum behavior, diverse rule base* col.2:4-20; *intelligent agents, rules, associated fitness, success measure* col.2:39-col.3:10; *FEHN, new rule, existing rule, effectiveness* col.3:45-50; *Engine 160, rules, survival of the fittest* col.7:20-36; *winning rule* col.11:60-67; *agent, previous best agent* col.26:35-46). It would have been obvious to one of ordinary skill in the pertinent art at the time the invention was made to incorporate the teaching of *Harrison et al.* into that of *Courts et al.* for the inclusion of champion/challenger experiment for testing a new strategy. And the motivation for doing so would have been to enable automatic and continual optimization or online adaptation (i.e., improving performance) of the software to minimize down time and sub-optimal functioning while awaiting new software version and avoid the ongoing needs to understand the software, modify tests and integrate the modified software (see at least col.2:20-36; col.4:17-33).

As per claims 34-38, 40-47, 49, 52, 57-61, 63-72, 75 they recite limitations which have

been addressed in claims 2, 4, 5, and 17, therefore, are rejected for the same reasons as cited in

claims 2, 4, 5, and 17.

9.   Claims 3 and 11 are rejected under 35 U.S.C. 103(a) as being unpatentable over *Courts et al.*

and *Harrison et al.* as applied to claims 1 and 9 above, and further in view of Dodrill et al. (US

6490564), hereinafter, *Dodrill et al..*

As per claim 3, *Courts et al.* teach the method as applied to claim 1 wherein software

components are implemented using programming languages such as J++, PB, VB, Delphi, or

C++ (e.g., *44* FIG.1 & associated text). *Courts et al.* do not expressly disclose generating code in

C. However, *Dodrill et al.* discloses a method and apparatus (e.g., see *application server 66*

FIG.3, 4 & associated text) for decision service returning a real-time decision (e.g., col.2:58-64) in

ASP mode to an end user (e.g., see *thin clients 42b & browser 56* FIG.4 & associated text, see

Abstract, col.2:44-51) wherein the user input (e.g., see *300 & 302* FIG.9 & associated text) is sent

to decision server via a Web server (e.g., see *Web Server 64* FIG. 4 & associated text) and

processed by the decision server according to installed rules (e.g., col.5:46-50), and

corresponding XML-formatted output data (e.g., see *dynamic HTML/XML pages 98* FIG.4 &

associated text) is generated and returned from decision server to Web server to be transmitted

to end user. *Dodrill et al.* further discloses a method and apparatus as described above wherein

applications/logic/functions/code (e.g., see *application 48* FIG.2 & associated text) are written in

programming language C (e.g., col.2:58-60) and formatted in CGI (e.g., col.2:61-63). It would

have been obvious to one of ordinary skill in the pertinent art at the time the invention was made

to substitute the programming languages disclosed in the teaching of *Courts et al.* with C to

produce the expected result with reasonable success. And the motivation for doing so would

have been the well-known characteristics/advantages associated with the C language, namely,

small size (i.e., few built-in functions) which allows flexibility and power in programming and

building/customizing the language for a specific application, portability (i.e., compiled on various

computer systems), and capacity for implementing system software/low-level tasks such as

transferring data and integrating system components, loading programs, and formatting text for

display, etc.,.

As per claim 11, it recites limitations which have been addressed in claims 9 & 3,

therefore, is rejected for the same reasons as cited in claims 9 & 3.

10.   Claims 6-7, 14-15 are rejected under 35 U.S.C. 103(a) as being unpatentable over *Courts et al.*

and *Harrison et al.* as applied to claims 1 and 9 above, and further in view of Humpleman et al.

(US 6466971), hereinafter, *Humpleman et al.*.

As per claim 6, *Courts et al.* teaches the method as applied to claim 1. *Courts et al.* do

not expressly disclose code generator server generating an XML schema for providing to a client

system for collecting said input data and said code generator server generating an XML

parser/builder for reading data conforming to said XML schema. However, *Humpleman et al.*

discloses a method and apparatus (e.g., see Abstract, FIG.14, 19 & associated text) for sending

XML input data (e.g., see *commands/XML* FIG.14 & associated text, see *XML-RPC Action*

FIG.19) from an end user/client system (e.g., see *A* FIG. 14 & associated text, see *HN Device A:*

Controller Module FIG.19 & associated text) to a decision server (e.g., see *S* FIG.14 & associated

text, see *HN Device B: Controller Module* FIG.19 & associated text) via a web server (e.g., see

server 14 FIG.14 & associated text, see *HN Device Web Server 86* FIG.19 & associated text),

said decision server processing the XML input data, generating XML-formatted response, web

pages and returning to the client via said web server (e.g., see *HTML or XML* FIG.14, see *XML-*

RPC Response FIG.19 & associated text). *Humpleman et al.* further discloses generating an

XML schema for providing to the client system for collecting said input data and providing to Web

server for use in error handling, or data validation (e.g., see *CALL.DTD & INTERFACE.DTD & Web Server Layer* FIG.18 & associated text, see *Device A XML Interface 72* FIG.19 & associated text) and generating an XML parser (e.g., see *XML Layer IN 70 & XML Layer OUT 68* FIG.18 & associated text, see *XML parser 74* FIG.19 & associated text) for reading data conforming to said XML schema. It would have been obvious to one of ordinary skill in the pertinent art at the time the invention was made to modify *Courts et al.*'s teaching to include the teaching as set forth by *Humpleman et al.* to produce the expected result with reasonable success. And the motivation for doing so would have been that the formatting of data into syntactically correct XML document(s) depends upon adhering to a predefined definition language describing the structure and set of constraints (i.e., XML schema) on which an XML documents shall be constructed from said data. Furthermore, XML parsers enable the processing and extracting of data in textual representation within XML tags and transforming them into specific-typed objects/data structure (e.g., C, C++, or Java objects) which can be retrieved for use by servers and software applications. Conventional XML parsers check XML documents being parsed for conformance to general XML rules. Most recent XML parsers, at the time the invention was made, are implemented with integrated support for XML schemas to further enable data validation.

As per claims 7, 14-15, they recite limitations which have been addressed in claims 1, 5, 6, therefore, are rejected for the same reasons as cited in claims 1, 5, 6.

11.   Claims 18-19, 23-27, 31-33, 39, 62 are rejected under 35 U.S.C. 103(a) as being unpatentable over *Courts et al.*, in view of *Harrison et al.* and further in view of Marullo et al. (*Marullo et al.,* US 6157940).

As per claim 18, *Courts et al.* teach a method and apparatus for an end user to develop rules, models, and/or strategies, for generating real time decisions in ASP mode (see claim 1), said method comprising:

- o   using a proprietary custom predictive analytics for outputting a models file of resulting rules by taking historical data as input (e.g., see *trend DB 36, profile DEB 38, trend data collection 32, business layer 16* FIG.1 & associated text);

- o   providing a designer component, said designer component providing means for designing rules, models, and strategies by using a project design (see claim 1);

- o   storing said project design in a projects repository for future reference (e.g., see *project database 148* FIG.2B & associated text);

- o   generating production code for executing in production mode (e.g., see claim 3).

Courts et al. disclose wherein said project design software further comprises capability for inserting an experiment for testing a new strategy (e.g., col.4:7-9; col.5:10-13; *business layer 16, business objects 22* FIG.1 & associated text). However, *Courts et al.* do not expressly disclose said experiment as champion/challenger experiment. However, *Harrison et al.* disclose a project design software (see at least *software development, solutions, learning classifier-system* col.1:25-60) comprises capability for inserting a champion/challenger experiment for testing a new strategy (see at least *learning classifier-system, rules, fitness* col.1:49-58; *genetic programming, single solution, learning classifier systems, multiple actions, knowledge base, optimum behavior, diverse rule base* col.2:4-20; *intelligent agents, rules, associated fitness, success measure* col.2:39-col.3:10; *FEHN, new rule, existing rule, effectiveness* col.3:45-50; *Engine 160, rules, survival of the fittest* col.7:20-36; *winning rule* col.11:60-67; *agent, previous best agent* col.26:35-46). It would have been obvious to one of ordinary skill in the pertinent art at the time the invention was made to incorporate the teaching of *Harrison et al.* into that of *Courts et al.* for the inclusion of champion/challenger experiment for testing a new strategy. And the motivation for doing so would have been to enable automatic and continual optimization or online adaptation (i.e., improving performance) of the software to minimize down time and sub-optimal functioning while awaiting

new software version and avoid the ongoing needs to understand the software, modify tests and integrate

the modified software (see at least col.2:20-36; col.4:17-33).

Courts et al do not expressly disclose generating a runtime version of said project design for testing, said

testing thereby validating and verifying said rules; stress testing said rules/models by inputting a

significantly large number of transactions into a monitor and Web server; said Web server generating a

bulk test report representing results of said stress testing; modifying said rules, models, and strategies, if

necessary as a result of said stress testing. However, *Marullo et al.* disclose an apparatus (e.g., see

FIG.2 & associated text, FIG.3 & associated text) and method of stress testing business/web-server

applications or functional areas within vertical markets (e.g., see Abstract, *commercial on-line banking*

and shopping transactions col.1:30-31, see *banking application 12* FIG.1 & associated text, see *banking*

application 32 FIG.2 & associated text), said apparatus and method comprising:

- o generating a runtime version of said project design and marking said project

  (e.g.,col.32:29-31) for testing (e.g., see *genautoAPI 58* FIG.2 & associated text, col.2:31-

  37, col.4:2-6, col.7:4:9 & 12-17), said testing thereby validating and verifying said rules

  (e.g., see FIG.13A, 13B & associated text, col.1:5-10, col.2:14-15 & 18-31, col.3:54-55 &

  60-65);

- o stress testing said rules/models (e.g., col.3:38-43, col.4:40-47, col.6:1-6 & 51-62, see

  webStrain 68 FIG.2 & associated text, see *352* FIG.18 & associated text, see FIG.16A-

  16C & associated text) by inputting a significantly large number of transactions into a

  monitor and Web server (e.g., see *web server 10* FIG.1 & associated text, col.1:43-47,

  see *genautoAPI 58* FIG.2 & associated text, see *106* FIG.15 & associated text);

- o said stress testing tracking and storing in repository (e.g., see *user specified files 40*

  FIG.3 & associated text) statistics on specific rules/models by counting the number of

  times predetermined rules/models are used during said stress testing (e.g., see *116, 118*

  FIG.9A & associated text, col.2:65-col.3:6,);

- o said Web server generating a bulk test report representing results of said stress testing (e.g., col.3:15-21, see *Reports 114* FIG.8 & associated text, see *360* FIG.18 & associated text);

- o modifying said rules/models if necessary as a result of said stress (e.g., col.3:38-43, col.4:40-47, col.6:1-6 & 51-62, see *webStrain 68* FIG.2 & associated text, col.1:5-10, col.2:14-15 & 18-31, col.3:54-55 & 60-65);

It would have been obvious to one of ordinary skill in the pertinent art at the time the invention was made to incorporate the teaching of *Marullo et al.* into that of *Courts et al.* and *Harrison et al.* (hereinafter *CHM*) to include the steps of stress testing rules/models as disclosed by *Marullo et al.* which would produce the expected result with reasonable success. And the motivation for doing so would have been that the automation of stress testing business/web-server applications (i.e., project design), verification/validation of rules/models, and report generation ensures that all possibilities of data input/output and all permutations and combinations of transactions/APIs and business logic/rules associated therewith have been exhaustively traversed, and tested for correctness and reported in a consistent, and efficient manner [in comparison to manual testing/traversing of links in web applications which yields unreliable test results not mirroring what is to be expected in the actual environment in which the web server applications would be used].

As per claims 19, 23-27, 31-33, 39, 62, they recite limitations which have been addressed in claims 3, 18, therefore, are rejected for the same reasons as cited in claims 3, 18.

12.   Claims 20, 28 are rejected under 35 U.S.C. 103(a) as being unpatentable over *CHM* as applied to claim 18 above and further in view of Ballantyne et al. (US 6687873), hereinafter, *Ballantyne et al.*.

As per claim 20, *CHM* teaches the method of claim 18. However *CHM* does not expressly disclose providing a model editor component for automatically converting said models file into an XML version of said rules, and importing said converted XML data into said designer component. However, *Ballantyne et al.* disclose a method (e.g., see abstract) and apparatus (e.g., see FIG.1 & associated text) of providing a model editor component (e.g., see *modeling engine 28, mapping engine 26, modeling/mapping GUI 30* FIG.1 & associated text) for automatically converting said rules/models file (e.g., see Abstract, see *legacy program applications 16* FIG.1 & associated text, see *36* FIG.2 & associated text) into an XML version of said rules/models (e.g., see *context table 22* FIG.1 & associated text, see *44* FIG.2 & associated text) and importing said converted XML data into said designer component (e.g., see *legacy system 12, writer engine 20* FIG.1 & associated text). It would have been obvious to one of ordinary skill in the pertinent art at the time the invention was made to incorporate the teaching of *Ballantyne et al.* into that of *CHM* to obtain a model editor component for automatically converting rules/models files in to XML format which are then imported to the designer component with reasonable success in producing the expected results. And the motivation for doing so would have been that automatic conversion of business rules/models into XML format eliminates the need to alter existing programming logic or business rules within legacy applications and further facilitates easy data transmission over the Internet, and between different applications, as well as direct display and manipulation of data via browser technology.

As per claim 28, it recites limitations, which have been addressed in claim 20, therefore, is rejected for the same reasons as cited in claim 20.

13.   Claims 21, 29, 50, 73 are rejected under 35 U.S.C. 103(a) as being unpatentable over *CHM* as applied to claim 18 above and further in view of Kendall et al. (US 2002/0138449), hereinafter, *Kendall et al.*.

As per claim 21, *CHM* teaches the method of claim 18. *CHM* does not expressly disclose said designer component further comprising providing designing software having graphical user interfaces for generating data, variables, rules, models, strategies, trees, and actions required in said project design. However, *Kendall et al.* disclose a method and apparatus (e.g., see FIG.1 & associated text) for providing a designing software having graphical user interfaces (GUIs) for generating data, variables, business rules/models, trees, and actions required in a project design (i.e., a visual designer component for facilitating said configuring said decision engine) (e.g., see Abstract, see FIG.5,6,9 & associated text). *Kendall et al.* further discloses generating for the project design a workflow functional component (e.g., see FIG.7,8 & associated text) having expression sequences (e.g., see *policy number, address, city, caller name* FIG.10 & associated text), segmentation trees (e.g., see *Driver is named on policy, police have been notified, injuries as a result of accident* FIG.5 & associated text), workflow lists (e.g., see FIG.5,9,10 & associated text) for means for placing said sequences, trees, and lists in a hierarchical order (e.g., (e.g., see FIG.5,9,10 & associated text) wherein a root workflow list (e.g., see *lost type is accident* FIG.5 & associated text) providing a starting point for processing workflow at runtime, and any of said workflow lists is used as a result list at an exit point of segmentation tree of said segmentation trees (e.g., see *outcome* FIG.6 & associated text), and wherein end result nodes of said segmentation tree points to said workflow list (e.g., see FIG.5,9,10 & associated text). It would have been obvious to one of ordinary skill in the pertinent art at the time the invention was made to incorporate the teaching of *Kendall et al.* into that of *CHM* to obtain a designing software having GUIs for generating data, variables, business rules/models, trees, and actions required in a project design. And the motivation for doing so would have been to enable the development and modification of evolving business logic/rules/models/actions by ordinary administrators/end-users without any computer programming experience and graphical displays of business rules/models/actions in forms of workflow lists, segmentation trees, and expression sequences further enable fast and easy analysis and/or modification of said rules/models/actions.

As per claims 29, 50, 73, they recite limitations, which have been addressed in claims 1,

and 18, 21, therefore, are rejected for the same reasons as cited in claims 1, 18, 21.

14.   Claims 22, 30 are rejected under 35 U.S.C. 103(a) as being unpatentable over *CHM* as applied to

claim 18 above and further in view of Bertrand et al. (US 6018732), hereinafter, *Bertrand et al.*.

As per claim 22, *CHM* teaches the method of claim 18, wherein rules are tested (see

claim 17). *CHM* does not expressly disclose providing a test service whereby said rules are

tested in runtime mode, said test service comprising a wrapper for a control panel and for an

Excel testing program. However, *Bertrand et al.* disclose a method and apparatus for returning

real-time decisions/scores/calculated results in ASP mode (e.g., see Abstract, see FIG.2 &

associated text), which is applicable to functional areas of vertical markets (e.g., see *domain

model* FIG.6 & associated text, see FIG.15, 16, 34, 75 & associated text, see col.21:15-32),

wherein rules are tested in runtime mode by a test service comprising a wrapper (e.g., see

presentation 210, activity 220 FIG.2 & associated text, see col.21:55-62, FIG.8 & associated text)

for a control panel and for an Excel testing program (i.e., a model editor for validating and

verifying content of rules/models) (e.g., see *simulation engine 270, simulation models 260* FIG.2

& associated text). *Bertrand et al* further disclose a model comprising an expert and decision

(e.g., see Abstract) wherein the model predicts revenue (e.g., col.11:23-32) and scores (e.g., see

interest rate, balance FIG.49 & associated text). It would have been obvious to one of ordinary

skill in the pertinent art at the time the invention was made to incorporate the teaching of *Bertrand

et al.* into that of *CHM* to obtain runtime test service comprising a wrapper for a the control panel

and for an Excel testing program. And the motivation for doing so would have been that the

usage of Excel spreadsheets in the test service/program enables business logic/rules/functions to

be collected, and simulated for testing purpose. Furthermore, Excel can be configured to enforce

data constraints and perform numerical calculations on data stored therein.

As per claim 30, it recites limitations, which have been addressed in claim 22, therefore, is rejected for the same reasons as cited in claim 22.

15.   Claims 51, 53-56, 74, 76-79 are rejected under 35 U.S.C. 103(a) as being unpatentable over *CHM* and *Kendall et al.* as applied to claim 50 above and further in view of *Humpleman et al.*.

As per claims 51, 53-56, 74, 76-79, they recite limitations, which have been addressed in claims 1, 6, 21, therefore, are rejected for the same reasons as cited in claims 1, 6, 21.

Conclusion

16.   Any inquiry concerning this communication or earlier communications from the examiner should be directed to Chrystine Pham whose telephone number is 571-272-3702. The examiner can normally be reached on Mon-Fri, 8:30am-5pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q. Dam can be reached on 571-272-3695. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

CP
April 15, 2005

TUAN DAM
SUPERVISORY PATENT EXAMINER